

EAST Search History

| Ref # | Hits | Search Query | DBs | Default Operator | Plurals | Time Stamp |
|-------|-------|---|---|------------------|---------|------------------|
| L1 | 95458 | (select\$4 or pick\$4 or choos\$4) with counter | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:54 |
| L2 | 167 | (select\$4 or pick\$4 or choos\$4) with (invert\$4 adj3 counter) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:53 |
| L3 | 2218 | 711/133-136,159-160.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:54 |
| L4 | 1 | I2 and I3 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:54 |
| L5 | 0 | (select\$4 or pick\$4 or choos\$4) with (line adj use adj counter) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:54 |
| S3 | 136 | ((least adj frequently adj used) same (most adj frequently adj used)) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:53 |
| S4 | 138 | ((least adj frequently adj used) or lfu) same ((most adj frequently adj used) or mfu) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 12:44 |
| S5 | 4 | S4 and 711/134.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/09/22 18:38 |

EAST Search History

| | | | | | | |
|-----|------|--|---|----|----|------------------|
| S6 | 42 | ((least adj frequently adj used) or lfu) same ((most adj frequently adj used) or mfu) and cache | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 09:25 |
| S7 | 2395 | (determin\$4 or calculat\$4 or predict\$4 or find\$4) near5 (new or next) adj2 state | US-PGPUB; USPAT | OR | ON | 2005/09/23 10:56 |
| S8 | 1 | S7 near7 (depend\$4 or based) near5 (cache near (line or page or row or bank)) | US-PGPUB; USPAT | OR | ON | 2005/09/23 10:57 |
| S9 | 2 | S7 same (depend\$4 or based) near5 (cache near (line or page or row or bank)) | US-PGPUB; USPAT | OR | ON | 2005/09/23 10:58 |
| S10 | 197 | 711/134.ccls. | US-PGPUB; USPAT | OR | ON | 2005/11/11 09:43 |
| S11 | 842 | two near2 cache near lines | US-PGPUB; USPAT | OR | ON | 2005/09/23 11:29 |
| S12 | 11 | S10 and S11 | US-PGPUB; USPAT | OR | ON | 2005/09/23 11:29 |
| S13 | 5 | ((least adj frequently adj used) or lfu) and ((most adj frequently adj used) or mfu) and 711/134.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 10:20 |
| S14 | 12 | ((least adj frequently adj used) or lfu) and ((most adj frequently adj used) or mfu)).ab. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 09:41 |
| S15 | 203 | 711/134.ccls. | US-PGPUB; USPAT | OR | ON | 2005/11/11 09:43 |
| S16 | 144 | (state adj information) near3 (cache adj lines) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 10:21 |
| S17 | 2 | ((least adj frequently adj used) and (most adj frequently adj used)).ti. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:45 |

EAST Search History

| | | | | | | |
|-----|-------|--|---|----|----|------------------|
| S18 | 2 | ((least adj frequently adj used) or lfu) and (mfu or(most adj frequently adj used))).ti. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 11:32 |
| S19 | 13 | ((least adj frequently adj used) or lfu).ti. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 11:35 |
| S20 | 21 | ((least adj frequently adj used) or lfu).ab. | US-PGPUB; USPAT | OR | ON | 2005/11/11 11:36 |
| S21 | 13 | "711"/\$.ccls. and S20 | US-PGPUB; USPAT | OR | ON | 2005/11/11 11:37 |
| S22 | 203 | 711/134.ccls. | US-PGPUB; USPAT | OR | ON | 2005/11/11 12:05 |
| S23 | 1939 | 711/133-136,159-160.ccls. | US-PGPUB; USPAT | OR | ON | 2006/07/21 08:45 |
| S24 | 1039 | (least adj frequently adj used) or lfu | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 12:06 |
| S25 | 12900 | (most adj frequently adj used) or mfu | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 12:06 |
| S26 | 85 | S23 and S24 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 12:06 |
| S27 | 58 | S23 and S25 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 12:17 |
| S28 | 19 | S26 and S27 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 12:06 |

EAST Search History

| | | | | | | |
|-----|-----|---|---|----|----|------------------|
| S30 | 10 | (select\$4 or choos\$4 or pick\$4) with ((least adj frequently adj used) or lfu) with ((most adj frequently adj used) or mfu) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 13:30 |
| S31 | 575 | (max or maximum or overflow or limit) near3 (counter or register) near5 (divid\$4 or invert\$4) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 13:32 |
| S32 | 1 | S31 and 711/133-136,159-160.ccls. | US-PGPUB; USPAT | OR | ON | 2005/11/11 13:31 |
| S33 | 1 | (max or maximum or overflow or limit) near3 (counter or register) near5 (divid\$4 or invert\$4) same cach\$4 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 13:32 |
| S34 | 14 | (max or maximum or overflow or limit) near3 (counter or register) near5 (divid\$4 or invert\$4) and cach\$4 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2005/11/11 13:32 |
| S35 | 1 | "20050114606" | US-PGPUB; USPAT | OR | ON | 2006/03/29 15:49 |
| S36 | 9 | (select\$4 or choos\$4 or pick\$4) with ((least adj frequently adj used) or lfu) with ((most adj frequently adj used) or mfu) | US-PGPUB; USPAT | OR | ON | 2006/03/29 18:10 |
| S37 | 1 | (select\$4 or choos\$4 or pick\$4) with ((least adj frequently adj used) or lfu) with ((most adj frequently adj used) or mfu) | USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/03/29 16:12 |
| S38 | 924 | (adjust\$4 or reset\$3 or resum\$3) near5 counter near5 threshold | US-PGPUB; USPAT | OR | ON | 2006/03/29 18:09 |
| S39 | 0 | S38 with ((most adj frequently adj used) or mfu) | US-PGPUB; USPAT | OR | ON | 2006/03/29 18:10 |
| S40 | 0 | S38 and ((most adj frequently adj used) or mfu) | US-PGPUB; USPAT | OR | ON | 2006/03/29 18:11 |
| S41 | 30 | S38 and "711"/\$.ccls. | US-PGPUB; USPAT | OR | ON | 2006/03/29 18:11 |

EAST Search History

| | | | | | | |
|-----|------|---|---|----|----|------------------|
| S44 | 128 | ((least adj frequently adj used) or lfu) with ((most adj frequently adj used) or mfu) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/06/11 12:36 |
| S45 | 13 | ((least adj frequently adj used) or lfu) with ((most adj frequently adj used) or mfu)).clm. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/06/11 13:09 |
| S48 | 33 | ("5586264" "5615362" "5737747" "5799185" "5960452" "6085193" "6351471" "6351474" "6594751" "20010003193" "20010034786" "20020029282" "20020097750" "20020116473" "20020162047" "20020181506" "20040039837" "5854887" "5918020" "6366970" "6463508" "6484212" "6505169" "6516361" "6593860" "6708213" "6742082" "6744763" "6665755" "6747991" "6760765" "20010052132" "20020056126").PN. | US-PGPUB; USPAT | OR | ON | 2006/06/11 13:10 |
| S49 | 11 | ("20020116585" "20030217113" "5043885" "6407680" "6721850" "20020078300" "20040049598" "20050053034" "5881266" "6272598" "6883066").PN. | US-PGPUB; USPAT | OR | ON | 2006/06/11 13:11 |
| S50 | 0 | ("2005/0114606").URPN. | USPAT | OR | ON | 2006/06/11 14:50 |
| S51 | 2218 | 711/133-136,159-160.ccls. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:45 |
| S52 | 13 | ((least adj frequently adj used) or lfu) and ((most adj frequently adj used) or mfu)).ab,ti. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:46 |
| S53 | 24 | ((least adj frequently adj used) or lfu) and ((most adj frequently adj used) or mfu)).clm. | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:48 |

EAST Search History

| | | | | | | |
|-----|-----|---|---|----|----|------------------|
| S54 | 4 | S52 and S53 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:48 |
| S55 | 13 | line near use near counter | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:48 |
| S56 | 223 | ((least adj frequently adj used) or lfu) and ((most adj frequently adj used) or mfu)) | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:48 |
| S57 | 0 | S55 and S56 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:48 |
| S58 | 0 | S51 and S55 | US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2006/07/21 08:48 |



[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide

cache and (choosing or selecting or picking) and (lfu or "least



THE ACM DIGITAL LIBRARY



[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

cache and choosing or selecting or picking and lfu or least frequently used and mfu or most frequently used and counter

Found 1,968 of 182,223

Sort results by

relevance



[Save results to a Binder](#)

[Try an Advanced Search](#)

[Try this search in The ACM Guide](#)

Display results

expanded form



[Search Tips](#)

☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Memory systems: Reducing energy consumption of queries in memory-resident](#)



[database systems](#)

Jayaprakash Pisharath, Alok Choudhary, Mahmut Kandemir

September 2004 **Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems**

Publisher: ACM Press

Full text available: [pdf\(177.88 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The tremendous growth of system memories has increased the capacities and capabilities of memory-resident embedded databases, yet current embedded databases need to be tuned in order to take advantage of new memory technologies. In this paper, we study the implications of hosting memory resident databases, and propose hardware and software (query-driven) techniques to improve their performance and energy consumption. We exploit the structured organization of memories, which enables a selective m ...

Keywords: DRAM, database, energy, hardware schemes, layouts, mapping, power consumption, query optimization, query-directed energy management

2 [A survey of Web cache replacement strategies](#)



Stefan Podlipnig, Laszlo Böszörményi

December 2003 **ACM Computing Surveys (CSUR)**, Volume 35 Issue 4

Publisher: ACM Press

Full text available: [pdf\(193.37 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Web caching is an important technique to scale the Internet. One important performance factor of Web caches is the replacement strategy. Due to specific characteristics of the World Wide Web, there exist a huge number of proposals for cache replacement. This article proposes a classification for these proposals that subsumes prior classifications. Using this classification, different proposals and their advantages and disadvantages are described. Furthermore, the article discusses the importance ...

Keywords: Web caching, replacement strategies

3 Research sessions: P2P and sensor networks: Compressing historical information in sensor networks 



Antonios Deligiannakis, Yannis Kotidis, Nick Roussopoulos

June 2004 **Proceedings of the 2004 ACM SIGMOD international conference on Management of data**

Publisher: ACM Press

Full text available:  pdf(172.89 KB) Additional Information: [full citation](#), [abstract](#), [references](#)

We are inevitably moving into a realm where small and inexpensive wireless devices would be seamlessly embedded in the physical world and form a wireless sensor network in order to perform complex monitoring and computational tasks. Such networks pose new challenges in data processing and dissemination because of the limited resources (processing, bandwidth, energy) that such devices possess. In this paper we propose a new technique for compressing multiple streams containing historical data fro ...



4 Area efficient architectures for information integrity in cache memories 



Seongwoo Kim, Arun K. Somani

May 1999 **ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture ISCA '99**, Volume 27 Issue 2

Publisher: IEEE Computer Society, ACM Press

Full text available:  pdf(227.09 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)
 [Publisher Site](#)


Information integrity in cache memories is a fundamental requirement for dependable computing. Conventional architectures for enhancing cache reliability using check codes make it difficult to trade between the level of data integrity and the chip area requirement. We focus on transient fault tolerance in primary cache memories and develop new architectural solutions, to maximize fault coverage when the budgeted silicon area is not sufficient for the conventional configuration of an error checki ...

5 A cache approach for supporting life-time UPT number 

Yi-Bing Lin

June 1996 **Wireless Networks**, Volume 2 Issue 2

Publisher: Kluwer Academic Publishers

Full text available:  pdf(570.33 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

A life-time Universal Personal Telecommunication (UPT) number allows a user to change the Personal Communications service provider without changing his/her UPT number. Since a translation from the life-time UPT number to a routing address must be done for every call placed, it is essential to make this process efficient. Two general approaches are the table lookup method and the database query method. The life-time UPT number can be easily implemented in the database query method. However, ...


6 Cache investment: integrating query optimization and distributed data placement 



Donald Kossmann, Michael J. Franklin, Gerhard Drasch, Wig Ag

December 2000 **ACM Transactions on Database Systems (TODS)**, Volume 25 Issue 4

Publisher: ACM Press

Full text available:  pdf(210.67 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Emerging distributed query-processing systems support flexible execution strategies in which each query can be run using a combination of data shipping and query shipping. As in any distributed environment, these systems can obtain tremendous performance and availability benefits by employing dynamic data caching. When flexible execution and dynamic caching are combined, however, a circular dependency arises: Caching occurs as

a by-product of query operator placement, but query operator pl ...

Keywords: cache investment, caching, client-server database systems, data shipping, dynamic data placement, query optimization, query shipping

7 A case for dynamic view management



Yannis Kotidis, Nick Roussopoulos

December 2001 **ACM Transactions on Database Systems (TODS)**, Volume 26 Issue 4

Publisher: ACM Press

Full text available: pdf(892.57 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Materialized aggregate views represent a set of redundant entities in a data warehouse that are frequently used to accelerate On-Line Analytical Processing (OLAP). Due to the complex structure of the data warehouse and the different profiles of the users who submit queries, there is need for tools that will automate and ease the view selection and management processes. In this article we present DynaMat, a system that manages dynamic collections of materialized aggregate views in a data warehouse ...

Keywords: Data cube, OLAP, data warehousing, materialized views

8 Fast detection of communication patterns in distributed executions



Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Publisher: IBM Press

Full text available: pdf(4.21 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

9 High performance data broadcasting systems



Peter Triantafillou, R. Harpantidou, M. Paterakis

August 2002 **Mobile Networks and Applications**, Volume 7 Issue 4

Publisher: Kluwer Academic Publishers

Full text available: pdf(382.65 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Data broadcasting as a means of efficient data dissemination is a key technology facilitating ubiquitous computing. For this reason, broadcast scheduling algorithms have received a lot of attention. However, all existing algorithms make the core assumption that the data items to be broadcast are immediately available in the transmitter's queue, ignoring the key role that the disk subsystem and the cache management play in the overall broadcast system performance. With this paper we contribute a ...

Keywords: broadcast scheduling, cache management, data dissemination, disk scheduling, system performance

10 Diverse topics: General adaptive replacement policies



Yannis Smaragdakis



October 2004 **Proceedings of the 4th international symposium on Memory management**

Publisher: ACM Press

Full text available: pdf(259.42 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

We propose a general scheme for creating adaptive replacement policies with good performance and strong theoretical guarantees. Specifically, we show how to combine any two existing replacement policies so that the resulting policy provably can never perform worse than either of the original policies by more than a small factor. To show that our scheme performs very well with real application data, we derive a virtual memory replacement policy that adapts between LRU, loop detection, LFU, and ...

Keywords: LRU, adaptive, replacement algorithms, virtual memory

11 Probing the black box: Transforming policies into mechanisms with infokernel



Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, Nathan C. Burnett, Timothy E. Denehy, Thomas J. Engle, Haryadi S. Gunawi, James A. Nugent, Florentina I. Popovici

October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**

Publisher: ACM Press

Full text available: pdf(365.12 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We describe an evolutionary path that allows operating systems to be used in a more flexible and appropriate manner by higher-level services. An infokernel exposes key pieces of information about its algorithms and internal state; thus, its default policies become mechanisms, which can be controlled from user-level. We have implemented two prototype infokernels based on the linuxtwofour and netbsdver kernels, called infolinux and infobsd, respectively. The infokernels export key abstractions as ...

Keywords: information, mechanism, policy

12 Principles of database buffer management



Wolfgang Effelsberg, Theo Haerder

December 1984 **ACM Transactions on Database Systems (TODS)**, Volume 9 Issue 4

Publisher: ACM Press

Full text available: pdf(2.39 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

This paper discusses the implementation of a database buffer manager as a component of a DBMS. The interface between calling components of higher system layers and the buffer manager is described; the principal differences between virtual memory paging and database buffer management are outlined; the notion of referencing versus addressing of database pages is introduced; and the concept of fixing pages in the buffer to prevent uncontrolled replacement is explained. Three basic t ...

13 Web services and performance evaluation: A fine-grained replacement strategy for XML query cache



Li Chen, Song Wang, Elizabeth Cash, Burke Ryder, Ian Hobbs, Elke A. Rundensteiner

November 2002 **Proceedings of the 4th international workshop on Web information and data management**

Publisher: ACM Press

Full text available: pdf(225.22 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Caching popular queries and reusing results of previously computed queries is one

important query optimization technique, especially in modern distributed environments such as the WWW. Based on the recent proliferation of XML data and the emergence of the XQuery language, we are thus developing a query- based caching system for XQuery queries, called ACE-XQ. ACE-XQ applies innovative query containment and rewriting strategies to answer incoming user queries based on the cached XQueries, whenever ...

Keywords: XML, XQuery, cache replacement strategy, query containment, query rewriting, semantic caching

14 Applications: Inferring relative popularity of internet applications by actively querying ☐

 DNS caches

Craig E. Wills, Mikhail Mikhailov, Hao Shang

October 2003 **Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement**

Publisher: ACM Press

Full text available:  pdf(257.56 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

In this work, we propose a novel methodology that can be used to assess the relative popularity for any Internet application based on the data servers it uses. The basic idea is to infer popularity of data servers by periodically "poking" at local Domain Name servers (LDNSs) that service Domain Name System requests from a set of users running Internet applications and determining if LDNSs have cached resource records for the data servers. This approach allows us to measure the relative percentag ...


Keywords: active content measurement, domain name system

15 Optimal File-Bundle Caching Algorithms for Data-Grids ☐

Ekow Otoo, Doron Rotem, Alexandru Romosan

November 2004 **Proceedings of the 2004 ACM/IEEE conference on Supercomputing**

Publisher: IEEE Computer Society

Full text available:  pdf(399.80 KB) Additional Information: [full citation](#), [abstract](#)

The file-bundle caching problem arises frequently in scientific applications where jobs process several files concurrently. Consider a host system in a data-grid that maintains a disk cache for servicing jobs of file requests where a job is serviced only if all its requested files are present in the disk cache. Files must now be admitted into the cache and replaced in sets of file-bundles. We show that traditional caching algorithms based on file popularity measures do not perform well since the ...

16 Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies ☐

Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, Prabhakar Raghavan

August 1998 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 7 Issue 3

Publisher: Springer-Verlag New York, Inc.

Full text available:  pdf(281.37 KB) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

We explore how to organize large text databases hierarchically by topic to aid better searching, browsing and filtering. Many corpora, such as internet directories, digital libraries, and patent databases are manually organized into topic hierarchies, also called *taxonomies*. Similar to indices for relational data, taxonomies make search and access more efficient. However, the exponential growth in the volume of on-line textual information makes it nearly impossible to maintain such taxono ...



Caching: Efficient prediction of web accesses on a proxy server



Wenwu Lou, Hongjun Lu

November 2002 **Proceedings of the eleventh international conference on Information and knowledge management**

Publisher: ACM Press

Full text available: pdf(350.55 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Web access prediction is an active research topic with many applications. Various approaches have been proposed for Web access prediction in the domain of individual Web servers but they have to be tailored to the domain of proxy servers to satisfy its special requirements in prediction efficiency and scalability. In this paper, the design and implementation of proxy-based prediction service (PPS) is presented. For prediction efficiency, PPS applies a new prediction scheme which employs a two-la ...

Keywords: navigational model, proxy server, web access prediction

18 DynaMat: a dynamic view management system for data warehouses



Yannis Kotidis, Nick Roussopoulos

June 1999 **ACM SIGMOD Record , Proceedings of the 1999 ACM SIGMOD international conference on Management of data SIGMOD '99**, Volume 28 Issue 2

Publisher: ACM Press

Full text available: pdf(1.44 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Pre-computation and materialization of views with aggregate functions is a common technique in Data Warehouses. Due to the complex structure of the warehouse and the different profiles of the users who submit queries, there is need for tools that will automate the selection and management of the materialized data. In this paper we present DynaMat, a system that dynamically materializes information at multiple levels of granularity in order to match the demand (workload) but also takes into ...

19 Performance evaluation of new adaptive object replacement techniques for VOD systems



B. Sonah, M. R. Ito

February 1999 **Proceedings of the 1999 ACM symposium on Applied computing**

Publisher: ACM Press

Full text available: pdf(619.65 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: VOD architecture, object replacement policies

20 Phase tracking and prediction



Timothy Sherwood, Suleyman Sair, Brad Calder

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture ISCA '03**, Volume 31 Issue 2

Publisher: ACM Press

Full text available: pdf(674.18 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

In a single second a modern processor can execute billions of instructions. Obtaining a bird's eye view of the behavior of a program at these speeds can be a difficult task when all that is available is cycle by cycle examination. In many programs, behavior is anything but steady state, and understanding the patterns of behavior, at run-time, can unlock a multitude of optimization opportunities. In this paper, we present a unified profiling

architecture that can efficiently capture, classify, and ...

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)